

Содержание

Графы	2
1 Задача А. Связность	2
2 Задача В. Дерево	3
Разные задачи	4
3 Задача С. Скобки	4
4 Задача D. Ядра	5

Графы

1 Задача А. Связность

Имя входного файла: `connect.in`
Имя выходного файла: `connect.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

В этой задаче требуется проверить, что граф является *связным*, то есть что из любой вершины можно по рёбрам этого графа попасть в любую другую.

Формат входных данных

В первой строке входного файла заданы числа N и M через пробел — количество вершин и рёбер в графе, соответственно ($1 \leq N \leq 100$, $0 \leq M \leq 10\,000$). Следующие M строк содержат по два числа u_i и v_i через пробел ($1 \leq u_i, v_i \leq N$); каждая такая строка означает, что в графе существует ребро между вершинами u_i и v_i .

Формат выходных данных

Выведите “YES”, если граф является связным, и “NO” в противном случае.

Примеры

<code>connect.in</code>	<code>connect.out</code>
3 2 1 2 3 2	YES
3 1 1 3	NO

2 Задача В. Дерево

Имя входного файла: `tree.in`
Имя выходного файла: `tree.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Дан неориентированный граф. Проверьте, является ли он деревом.

Формат входных данных

В первой строке входного файла заданы через пробел два целых числа n и m — количество вершин и рёбер в графе, соответственно ($1 \leq n \leq 100$). В следующих m строках заданы рёбра; i -я из этих строк содержит два целых числа u_i и v_i через пробел — номера концов i -го ребра ($1 \leq u_i, v_i \leq n$). Граф не содержит петель и кратных рёбер.

Формат выходных данных

В первой строке выходного файла выведите “YES”, если граф является деревом, и “NO” в противном случае.

Примеры

<code>tree.in</code>	<code>tree.out</code>
3 2 1 2 1 3	YES
3 3 1 2 2 3 3 1	NO

Разные задачи

3 Задача С. Скобки

Имя входного файла: `brackets.in`
Имя выходного файла: `brackets.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 мегабайт

Правильная скобочная последовательность (ПСП) определяется следующим образом:

1. Пустая строка является ПСП.
2. Если S — ПСП, то (S) также является ПСП.
3. Если A и B являются ПСП, то и AB (конкатенация A и B) также является ПСП.

Напишите программу, выводящую все ПСП заданной длины.

Формат входных данных

В единственной строке входного файла записано целое число n . $1 \leq n \leq 10$.

Формат выходных данных

Выведите по одной строке для каждой ПСП длины $2n$ (из n пар скобок). Порядок не имеет значения.

Пример

<code>brackets.in</code>	<code>brackets.out</code>
1	()
2	()() (())

4 Задача D. Ядра

Имя входного файла: `balls.in`
Имя выходного файла: `balls.out`
Ограничение по времени: 1 секунда
Ограничение по памяти: 256 Мебибайт

Капитан Вася всегда держит на своем корабле запас пушечных ядер для борьбы с пиратами. Так как он привык во всем поддерживать порядок, он хранит ядра в виде пирамид. Каждый из слоев одной пирамиды является равносторонним заполненным ядрами треугольником, сторона которого содержит ровно k ядер. Сторона основания пирамиды состоит из n ядер, в следующем слое сторона состоит из $n - 1$ ядра, и т.д., пока на вершину не будет положено одно ядро (которое является равносторонним треугольником со стороной 1).

Например, пирамида размера 3 состоит из трех уровней, выглядящих так (сверху вниз):

```
  X
  X
 X X

  X
 X X
X X X
```

Ясно, что каждый из треугольников может содержать только 1, 3, 6, 10 и т.д. ядер. Таким образом, пирамида может содержать только 1, 4, 10, 20, и т.д. ядер.

Вася отправляется в плавание и берет с собой ровно m ядер. Какое минимальное число пирамид требуется ему сложить из них на своем корабле?

Формат входных данных

В первой строке входного файла записано количество тестов $1 \leq T \leq 20$. В последующих T строках задается количество ядер в i -м тесте $1 \leq m_i \leq 300\,000$.

Формат выходных данных

Для каждого из T тестов входного файла выведите в отдельной строке минимальное количество пирамид.

Пример

<code>balls.in</code>	<code>balls.out</code>
5	1
1	2
5	3
9	3
15	2
91	